

RANCANG BANGUN SIMULASI LANGKAH KUDA DALAM PAPAN CATUR

T U G A S A K H I R

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Pada
Jurusan Teknik Informatika

Oleh :

AMRI SAPUTRA
10451025499



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM
RIAU
PEKANBARU
2011**

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Assalamu'alaikum Wr. Wb.

Alhamdulillah Rabbil Alamin penulis ucapkan sebagai tanda syukur yang tak terhingga kepada Allah SWT atas segala karunia dan rahmat yang diberikan-Nya, sehingga penulis dapat melaksanakan dan akhirnya dapat menyelesaikan Tugas akhir ini. Shalawat beriring salam terucap buat junjungan alam Rasulullah Muhammad SAW, yang telah membawa perubahan yang sangat berarti.

Tugas akhir ini disusun sebagai syarat kelulusan pada jurusan Teknik Informatika Universitas Islam Negeri Sultan Syarif Kasim Pekanbaru.

Banyak sekali pihak yang telah membantu penulis dalam menyusun Tugas akhir ini, baik berupa materi maupun berupa moril/motivasi. Untuk itu pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. H. M. Nazir, selaku Rektor Universitas Islam Negeri Sultan Syarif Kasim Riau.
2. Ibu Dra. Hj. Yenita Morena, M.Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau.
3. Bapak Novriyanto, ST, M.Sc selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Sultan Syarif Kasim Riau.
4. Bapak Febi Yanto, M.Kom selaku penguji Tugas akhir, yang telah memberi bimbingan, arahan, dan saran yang berharga dalam menyusun Tugas akhir ini.
5. Ibu Fitra Kurnia, S.Kom selaku penguji Tugas akhir, yang telah memberi bimbingan serta saran yang berharga dalam menyusun Tugas akhir ini.

6. Ibu Elin Haerani, ST, M.Kom selaku pembimbing Tugas akhir, yang telah memberi bimbingan, arahan, dan saran yang berharga dalam menyusun Tugas akhir ini.
7. Bapak Suwanto Sanjaya, ST selaku Koordinator Tugas akhir, yang telah memberi arahan dan saran dalam menyusun Tugas akhir ini.
8. Seluruh staff dosen dan karyawan Fakultas Sains dan Teknologi, khususnya Jurusan Teknik Informatika UIN SUSKA Pekanbaru.
9. Ayahku Nazirman Anas dan Ibuku Syamsiah yang tercinta. Terima kasih atas segala pengorbanan dan cucuran keringatmu, terimakasih atas kasih sayang tak terhingga yang selalu menyirami hidupku, terimakasih atas doa yang slalu menyertai langkahku. Ya Allah, lindungi dan sayangilah mereka sebagaimana mereka menyayangiku sewaktu aku kecil hingga sekarang dan sampai aku mati nanti dan penulis mintakan syurga atas segala pengorbanannya.
10. Semua kakakku dan abang-abangku yang selalu memberikan semangat, arahan, dorongan dalam belajar.
11. Istriku tercinta dan terkasih dan My baby fadhila balqiz yang lucu dan maniz yang telah memberikan semangat, motivasi, dorongan dalam menyelesaikan Tugas akhir. Kebersamaan, do'a, kasih sayang, perhatian, motivasi, semangat dan dukungan yang selalu mendorongku untuk terus maju. Maafkan jika aku belum bisa menjadi yang terbaik.
12. Teman-teman seperjuangan di Jurusan Teknik Informatika angkatan 2004, khususnya di Base Camp terima kasih atas saran, kritik dan diskusi yang sangat membangun.
13. Dan pihak lain yang tidak dapat disebutkan satu persatu

Tidak ada yang sempurna, karena kesempurnaan hanya milik Allah semata. Tidak ada yang sia-sia dari apa yang kita usahakan. Terima kasih penulis ucapkan dan mohon maaf jika ada kekurangan, kesilapan dan kesalahan dalam penulisan tugas akhir ini. Dari itu semua, penulis membuka diri dalam menerima masukan berupa

kritik dan saran yang membangun dari semua pihak untuk penyempurnaan dan agar dapat lebih baik di masa yang akan datang.

Dan akhir kata penulis berharap semoga laporan ini dapat bermanfaat bagi semua pihak, khususnya dalam bidang pendidikan Teknik Informatika.

Wassalam

Pekanbaru, 22 Juni 2011

Penulis

RANCANG BANGUN SIMULASI LANGKAH KUDA DALAM PAPAN CATUR

AMRI SAPUTRA
10451025499

Tanggal Sidang : 22 Juni 2011
Periode Wisuda : November 2011

Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Sultan Syarif Kasim Riau
Jl. Soebrantas KM 15 No. 155 Pekanbaru

ABSTRAK

Catur merupakan permainan mental yang dimainkan oleh dua orang. Pergerakan langkah biji kuda dalam papan catur merupakan masalah klasik dalam *Artificial Intelligence*. Biji kuda dalam papan catur mempunyai pergerakan langkah yang unik. Maka dari itu dibuatlah suatu aplikasi yang dapat mensimulasikan pergerakan langkah kuda dalam papan catur untuk menggambarkan pergerakan langkah kuda dari posisi awal menuju posisi tujuan. Aplikasi simulasi ini diimplementasikan menggunakan bahasa pemrograman *Visual Basic 6.0* dengan sistem pencarian menggunakan metode *Breadth First Search (BFS)*. Aplikasi ini mempunyai kemampuan untuk mencari semua solusi pergerakan terpendek (*shortest path*) dari posisi awal menuju ke suatu posisi tujuan. Aplikasi ini mengimplementasikan konsep *Artificial Intelligence* dalam mencari solusi khususnya pohon pencarian.

Kata kunci : *Artificial Intelligence, Breadth First Search, Simulasi*

DAFTAR ISI

	Halaman
LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN	iii
LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL	iv
LEMBAR PERNYATAAN	v
LEMBAR PERSEMBAHAN	vi
ABSTRAK	vii
<i>ABSTRACT</i>	viii
KATA PENGANTAR	ix
DAFTAR ISI	xii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
DAFTAR ISTILAH	xviii
 BAB I. PENDAHULUAN	 I-1
1.1. Latar belakang	I-1
1.2. Rumusan masalah	I-2
1.3. Batasan masalah	I-2
1.4. Tujuan penulisan	I-3
1.5. Sistematika penulisan	I-3
 BAB II. LANDASAN TEORI	 II-1
2.1. AI (Artificial Intelligence)	II-1
2.1.1 Sejarah AI	II-1
2.1.2 Defenisi AI	II-2
2.2. Sistem Produksi.....	II-3
2.3 Masalah Ruang Keadaan (State Space Problem)	II-4
2.4 Metode pencarian (BFS)	II-5

2.4.1 Pencarian Buta	II-5
2.4.1.1 Pencarian Melebar Pertama	II-5
2.4.1.1.1 Cara Kerja Algoritma BFS	II-6
2.4.1.1.2 Aplikasi Algoritma BFS Teori Graf	II-8
2.4.1.2 Pencarian Mendalam Pertama	II-9
2.5 Asal-usul permainan catur	II-10
2.5.1 Catur	II-10
2.5.2 Gerakan	II-10
2.5.3 Rokade	II-12
2.5.4 Skak	II-12
2.5.5 Akhir Permainan	II-12
2.6 Permasalahan Langkah Kuda Dalam Papan Catur	II-13
BAB III METODOLOGI PENELITIAN	III-1
3.1. Pembelajaran Secara Literatur	III-1
3.2 Pengumpulan Data	III-2
3.3 Analisa Dan Perancangan	III-2
3.4 Implementasi dan Pengujian	III-3
3.5 Kesimpulan dan Saran	III-3
BAB IV ANALISA DAN PERANCANGAN	IV-1
4.1 Analisa	IV-1
4.1.1 Algoritma Pencarian Solusi	IV-5
4.1.2 Algoritma Fungsi Pendukung	IV-6
4.2 Perancangan	IV-9
4.2.1. Form Input	IV-9
4.2.2. Form Solusi	IV-11
4.2.3. Form About	IV-12
BAB V IMPLEMENTASI DAN PENGUJIAN	V-1
5.1. Implementasi	V-1
5.1.1. Lingkungan implementasi	V-1

5.1.1.1 Perangkat Keras	V-1
5.1.1.2 Perangkat Lunak.....	V-1
5.1.2. Hasil implementasi	V-1
5.2. Pengujian Perangkat Lunak (Testing).....	V-6
5.2.1 Pengujian Tampilan	V-6
5.2.1.1 Pengujian Halaman Input.....	V-6
5.2.1.2 Pengujian Halaman Simulasi	V-8
5.2.1.3 Pengujian Halaman About	V-9
5.3 Kesimpulan Pengujian	V-10
BAB VI. PENUTUP	VI-1
6.1. Kesimpulan	VI-1
6.2. Saran	VI-1
DAFTAR PUSTAKA	
DAFTAR RIWAYAT HIDUP	

DAFTAR ISTILAH

<i>Blackbox</i>	Pengujian Untuk Membuktikan Proses Berjalan Dengan Baik (Tidak <i>Error</i>)
<i>Form</i>	Bentuk dari sebuah tampilan
<i>Goal state</i>	Keadaan Tujuan
<i>Implementasi</i>	Pelaksanaan atau penerapan
<i>Initial state</i>	Keadaan Awal
<i>Input</i>	Data yang dimasukkan
<i>Interface</i>	Tampilan antar muka
<i>Output</i>	Data yang dihasilkan
Proses	Unit kerja terkecil yang secara individu memiliki sumber daya-sumber daya dan dijadwalkan sistem operasi
Simulasi	Proses merancang model dari suatu sistem yang sebenarnya,

DAFTAR TABEL

Tabel	Halaman
4.1 Solusi Permasalahan Langkah Kuda	IV-9
4.2 Rancangan <i>Form Input</i>	IV-10
4.3 Rancangan <i>Form</i> solusi	IV-11
4.4 Rancangan <i>Form About</i>	IV-12
5.1 Solusi Yang Didapatkan (contoh kasus 1)	V-4
5.2 Solusi Yang Didapatkan (contoh kasus 2)	V-5
5.3 Pengujian Halaman Input	V-6
5.4 Pengujian Halaman Simulasi	V-8
5.5 Pengujian Halaman About	V-9

DAFTAR GAMBAR

Gambar	Halaman
2.1 Penerapan konsep AI di dalam komputer	II-3
2.2 Sistem Produksi	II-3
2.3 Metode Breadth First Search.....	II-6
2.4 Pohon Permasalahan	II-6
2.5 Pohon Rute Permasalahan	II-7
2.6 Graf Berarah	II-8
2.7 Graf Tidak Berarah	II-8
2.8 Contoh Kasus Graf Tidak Berarah	II-8
2.9 Metode Depth First Search	II-9
2.10 Gambar Papan catur 8 X 8.....	II-14
3.1 Diagram Langkah-Langkah Penelitian	III-1
4.1 Simbol Posisi Awal	IV-1
4.2 Simbol Posisi Tujuan	IV-1
4.3 Simbol Posisi Penghalang	IV-1
4.4 Papan Catur 8 X 8.....	IV-2
4.5 Aturan Pergerakan-1	IV-2
4.6 Aturan Pergerakan-2.....	IV-3
4.7 Aturan Pergerakan-3	IV-3
4.8 Aturan Pergerakan-4.....	IV-3
4.9 Aturan Pergerakan-5	IV-4
4.10 Aturan Pergerakan-6.....	IV-4
4.11 Aturan Pergerakan-7.....	IV-4
4.12 Aturan Pergerakan-8.....	IV-5
4.13 Posisi Dalam Papan Catur	IV-6
4.14 Contoh Permasalahan Kuda.....	IV-7

4.15	Pencarian BFS Pada Problema Kuda.....	IV-8
4.16	Rancangan Form Input	IV-10
4.17	Rancangan Form Solusi.....	IV-11
4.18	Rancangan Form About.....	IV-12
5.1	Form Input Peringatan Untuk Posisi Awal.....	V-2
5.2	Form Input Peringatan Untuk Posisi Tujuan	V-2
5.3	Form Input (Contoh Kasus 1).....	V-3
5.4	Form Solusi (Contoh Kasus 1)	V-3
5.5	Form Input (Contoh Kasus 2).....	V-4
5.6	form solusi (contoh kasus 2).....	V-5
5.7	Form Simulasi Pergerakan Langkah kuda.....	V-5

BAB I

PENDAHULUAN

1.1 Latar Belakang

Catur adalah permainan mental yang dimainkan oleh dua orang. Pecatur adalah orang yang memainkan catur, baik dalam pertandingan satu lawan satu maupun satu melawan banyak orang (dalam keadaan informal). Kata catur diambil dari bahasa Sanskerta yang berarti "empat", namun kata ini sebenarnya merupakan singkatan dari *caturangga* yang berarti empat sudut. Di India kuno permainan catur memang dimainkan oleh empat peserta yang berada di empat sudut yang berbeda. Hal ini lain dari permainan catur modern di mana pesertanya hanya dua orang saja, kemudian kata *caturangga* ini diserap dalam bahasa Persia menjadi *shatranj*. Kata *chess* dalam bahasa Inggris diambil dari bahasa Persia *shah*. (Bey Magethi. 2009)

Biji catur terdiri dari Raja, Ratu, Gajah/Rencong, Kuda, Benteng dan Pion. Dari sekian banyak biji catur, biji catur kuda mempunyai langkah yang lain dari biji catur lainnya. Biji catur kuda dalam papan catur memiliki pergerakan menyerupai huruf L dan juga dapat melangkah walaupun terdapat bidak/pion di depannya. Biji catur ini merupakan salah satu biji yang sangat sulit digerakkan dan sering juga merupakan biji yang paling berbahaya apabila tidak diperhatikan secara seksama setiap pergerakannya. Simulasi dari permasalahan ini menyediakan sebuah papan catur berukuran 8 x 8. Sasaran (*goal*) dari permasalahan ini adalah menggerakkan sebuah biji kuda dari suatu posisi tertentu pada papan catur ke posisi tujuan yang diinginkan dengan mensimulasikan semua solusi pergerakan terpendek yang mungkin untuk menuju ke posisi tujuan tersebut. Permasalahan ini juga merupakan salah satu masalah klasik dalam *artificial intelligence* (AI). Berdasarkan Tugas akhir Ira Rasyidah yang berjudul Analisa Perbandingan Algoritma *Breadth First Search* (BFS) dan Algoritma *Least Cost Search* (LCS) studi kasus: Game of 8, mengatakan pada *optimality* algoritma *Breadth First Search* pada pencarian solusi game of 8 dapat

menemukan semua solusi yang optimal. Oleh karena itu, penulis mencoba menyelesaikan permasalahan ini menggunakan metode pencarian melebar pertama (*Breadth First Search*) dan bantuan sistem produksi.

Hubungan permasalahan langkah kuda dalam papan catur dengan sistem produksi adalah:

1. Posisi awal kuda sebagai keadaan awal.
2. Posisi tujuan kuda adalah Raja.
3. Pergerakan kuda yang menyerupai huruf L sebagai aturan.
4. Metode pencarian *Breadth First Search*, sebagai strategi kontrol.

Berdasarkan uraian di atas, penulis ingin merancang suatu aplikasi yang mampu mencari semua solusi pergerakan terpendek sebuah biji kuda dari posisi awal menuju posisi tujuan yang bersifat dinamis dan menampilkan simulasi pergerakan dari kuda dalam papan catur. Oleh karena itu, penulis mengambil tugas akhir dengan judul **“Rancang Bangun Simulasi Langkah Kuda dalam Papan Catur”**

1.2 Rumusan Masalah

Adapun masalah yang akan dibahas dalam penelitian untuk penyusunan tugas akhir ini yaitu :

- a. Mencari dan menampilkan semua solusi pergerakan terpendek yang dapat dilewati biji kuda dari posisi awal ke posisi tujuan.
- b. Mensimulasikan semua solusi pergerakan yang dihasilkan.

1.3 Batasan Masalah

Karena keterbatasan waktu dan pengetahuan penulis, maka ruang lingkup permasalahan dalam merancang aplikasi ini antara lain :

1. Ukuran dari papan catur 8 x 8
2. Komponen-komponen yang terdapat pada aplikasi, yaitu:
 - Kuda (putih), sebagai simbol dari posisi awal.
 - Raja (hitam), sebagai simbol dari posisi tujuan.
 - Bidak/Pion (hitam), sebagai simbol dari rintangan yang tidak boleh ditempati. Bidak / rintangan dibatasi maksimal sebanyak 8 buah.

3. Metode pencarian yang digunakan adalah dengan menggunakan metode pencarian melebar pertama (*Breadth-First Search*).
4. Pencarian hanya akan menemukan semua solusi pergerakan terpendek.
5. Aplikasi yang akan dirancang bukan suatu permainan/ *game*, hanya memperlihatkan langkah kuda dari posisi awal hingga mencapai posisi tujuan (raja)

1.4 Tujuan Penelitian

Tujuan penyusunan tugas akhir ini adalah untuk merancang suatu aplikasi yang mampu mencari semua solusi pergerakan terpendek untuk menuju posisi tujuan dengan menggunakan bantuan sistem produksi dan menggunakan metode BFS (*Breadth-First Search*)

1.5 Sistematika Penulisan

Sistematika yang disajikan dalam tugas akhir ini meliputi enam bab yang susunannya adalah sebagai berikut :

- BAB I : Pendahuluan
 Pada bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan , dan sistematika penulisan tugas akhir.
- BAB II : Landasan Teori
 Pada bab ini berisikan tentang landasan teori yang mendukung pembuatan aplikasi yang meliputi pengetahuan tentang *Artificial Intelligence* (AI), metode BFS (*Breadth First Search*) dan Sistem Produksi
- BAB III : Metode Penelitian
 Berisi tentang metodologi atau urutan, tata cara dan langkah-langkah penelitian guna penunjang dalam pembuatan aplikasi.
- BAB IV : Analisa Dan Perancangan Aplikasi
 Pada bab ini berisi perancangan dan proses pembuatan tugas akhir.
- BAB V : Implementasi
 Pada bab ini berisi tentang hasil pengujian yang telah dilaksanakan dan melakukan analisis terhadap hasil dari ujicoba.

BAB VI : Penutup

Pada bab ini berisi kesimpulan yang didapat dari hasil pengujian dan juga saran untuk pengembangan lebih lanjut.

BAB II

LANDASAN TEORI

2.1 Artificial Intelligence (AI)

2.1.1 Sejarah AI

Kecerdasan buatan atau AI (*Artificial Intelligence*) termasuk bidang ilmu yang relatif muda. Pada tahun 1950-an para ilmuwan dan peneliti mulai memikirkan bagaimana caranya agar mesin dapat melakukan pekerjaannya seperti yang bisa dikerjakan oleh manusia. Alan Turing, seorang matematikawan dari Inggris pertama kali mengusulkan adanya tes untuk melihat bisa tidaknya sebuah mesin dikatakan cerdas. Hasil tes tersebut kemudian dikenal dengan Turing Test, dimana si mesin tersebut menyamar seolah-olah sebagai seseorang di dalam suatu permainan yang mampu memberikan respon terhadap serangkaian pertanyaan yang diajukan. Turing beranggapan bahwa, jika mesin dapat membuat seseorang percaya bahwa dirinya mampu berkomunikasi dengan orang lain, maka dapat dikatakan bahwa mesin tersebut cerdas, seperti layaknya manusia. (Kusumadewi.S, 2002)

AI itu sendiri dimunculkan oleh seorang profesor dari Massachusetts Institute of Technology yang bernama John McCarthy pada tahun 1956 pada Dartmouth Conference yang dihadiri oleh para peneliti AI. Pada konferensi tersebut juga didefinisikan tujuan utama dari kecerdasan buatan, yaitu mengetahui dan memodelkan proses-proses berpikir manusia dan mendesain mesin agar dapat meniru kelakuan mesin tersebut. Beberapa program AI yang mulai dibuat pada tahun 1956-1966, antara lain:

1. Logic Theorist, diperkenalkan pada Dartmouth Conference, program ini dapat membuktikan teorema-teorema matematika.
2. Sad Sam, diprogram oleh Robert K. Lindsay (1960). Program ini dapat mengetahui kalimat-kalimat sederhana yang ditulis dalam bahasa Inggris dan mampu memberikan jawaban dari fakta-fakta yang didengar dalam sebuah percakapan.

3. ELIZA, diprogram oleh Joseph Weizenbaum (1967). Program ini mampu melakukan terapi terhadap pasien dengan memberikan beberapa pertanyaan dan jawaban. (Kusumadewi.S, 2002)

2.1.2 Definisi AI

Kecerdasan buatan atau AI adalah salah satu bagian ilmu komputer yang mempelajari tentang bagaimana cara membuat agar komputer dapat melakukan pekerjaan seperti yang dilakukan oleh manusia. Tujuan dari AI adalah untuk memecahkan persoalan dunia nyata (bersifat praktis) dan memahami inteligensi (bersifat memahami). (Sandi.S, 1993)

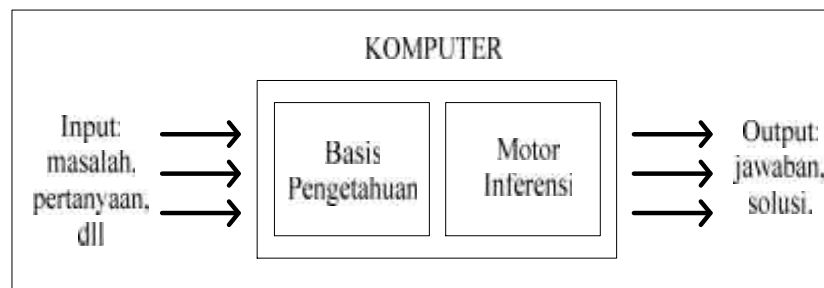
AI merupakan salah satu bagian ilmu komputer yang mempelajari tentang bagaimana cara membuat agar komputer dapat melakukan pekerjaan seperti yang dilakukan oleh manusia. Pada awal diciptakannya, komputer hanya difungsikan sebagai alat hitung saja. Namun seiring dengan perkembangan zaman, maka peran komputer semakin mendominasi kehidupan umat manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, lebih dari itu, komputer diharapkan untuk dapat diberdayakan untuk mengerjakan segala sesuatu yang bisa dikerjakan oleh manusia. (Kusumadewi.S, 2002)

Manusia bisa menjadi pandai dalam menyelesaikan segala permasalahan di dunia ini karena manusia mempunyai pengetahuan dan pengalaman. Pengetahuan diperoleh dari belajar. Semakin banyak bekal pengetahuan yang dimiliki oleh seseorang tentu saja diharapkan akan lebih mampu dalam menyelesaikan permasalahan, namun bekal pengetahuan saja tidak cukup, manusia juga diberi akal untuk melakukan penalaran, mengambil kesimpulan berdasarkan pengetahuan dan pengalaman yang mereka miliki. Tanpa memiliki kemampuan menalar yang baik, manusia dengan segudang pengalaman dan pengetahuan tidak akan dapat menyelesaikan masalah dengan baik. Demikian pula, dengan kemampuan menalar yang sangat baik, namun tanpa bekal pengetahuan dan pengalaman yang memadai, manusia juga tidak akan bisa menyelesaikan masalah dengan baik. (Kusumadewi.S, 2002)

Agar komputer bisa bertindak seperti dan sebaik manusia, maka komputer juga harus diberi bekal pengetahuan dan mempunyai kemampuan untuk menalar.

Untuk itu pada *Artificial Intelligence*, akan mencoba untuk memberikan beberapa metode untuk membekali komputer dengan kedua komponen tersebut (pengetahuan dan menalar) agar komputer bisa menjadi mesin yang pintar. (Kusumadewi.S, 2002). Untuk menciptakan aplikasi AI ada 2 bagian utama yang sangat dibutuhkan, yaitu:

1. Basis Pengetahuan (*Knowledge Base*), berisi fakta-fakta, teori, pemikiran dan hubungan antara satu dengan yang lainnya.
2. Motor Inferensi (*Inference Engine*), yaitu kemampuan menarik kesimpulan berdasarkan pengalaman. (Kusumadewi.S, 2002)

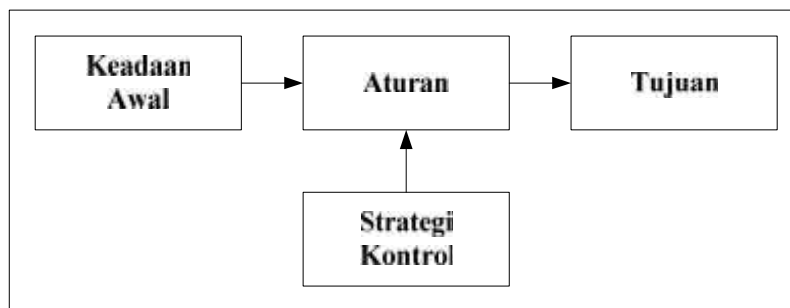


Gambar 2.1 Penerapan konsep AI di dalam komputer

2.2 Sistem Produksi

Sistem produksi terdiri dari komponen-komponen sebagai berikut:

1. Ruang keadaan, yang berisi keadaan awal, tujuan dan sekumpulan aturan yang digunakan untuk mencapai tujuan.
2. Strategi Kontrol, yang berguna untuk mengarahkan bagaimana proses pencarian akan berlangsung dan mengendalikan arah eksplorasi



Gambar 2.2 Sistem Produksi

Sistem Produksi merupakan salah satu bentuk representasi pengetahuan yang sangat populer dan banyak digunakan. Representasi pengetahuan dengan sistem produksi, pada dasarnya berupa aplikasi aturan (*rule*) yang berupa:

1. *Antecedent*, yaitu bagian yang mengekspresikan situasi atau premis.
2. Konsekuensi, yaitu bagian yang menyatakan suatu tindakan tertentu atau konklusi yang diterapkan jika suatu situasi atau premis bernilai benar.

Konsekuensi atau konklusi baru dinyatakan benar, jika bagian *antecedent* pada sistem tersebut juga benar atau sesuai dengan aturan tertentu. Apabila pengetahuan direpresentasikan dengan aturan, maka ada 2 metode penalaran yang dapat digunakan yaitu :

1. *Forward Reasoning* (penalaran maju). Penalaran dimulai dari keadaan awal dan kemudian dicoba untuk mencocokkan dengan tujuan yang diharapkan.
2. *Backward Reasoning* (penalaran mundur). Penalaran ini dimulai dari tujuan atau hipotesa, baru dicocokkan dengan keadaan awal atau fakta-fakta yang ada.

Hubungan permasalahan langkah kuda dalam papan catur dengan sistem produksi adalah:

1. Posisi awal kuda sebagai keadaan awal.
2. Posisi tujuan kuda sebagai keadaan tujuan.
3. Pergerakan kuda yang menyerupai huruf L sebagai aturan.
4. Metode pencarian *Breadth First Search*, sebagai strategi kontrol.

2.3 Masalah Ruang Keadaan (*State Space Problem*)

State merupakan representasi suatu keadaan pada suatu saat ataupun deskripsi konfigurasi sistem. *State space* adalah semua *state* yang mungkin dan biasanya digambarkan sebagai jaringan dengan *verteks* merupakan *state* dan *edge* merupakan perubahan yang mungkin. Deskripsi formal suatu masalah :

1. Definisikan *state space*.
2. Tentukan *initial state* dalam *goal state*.
3. Tentukan operator-operator.

Representasi *state space* memungkinkan definisi formal suatu masalah sebagai persoalan mengubah status dengan menggunakan sekumpulan operator (*rule*) dan juga mendefinisikan masalah sebagai *search* yaitu mencari lintasan di dalam *state space* dari *initial state* ke *goal state*. Contohnya : permasalahan langkah kuda dalam papan catur. (Kusumadewi.S, 2002)

2.4 Metode Pencarian

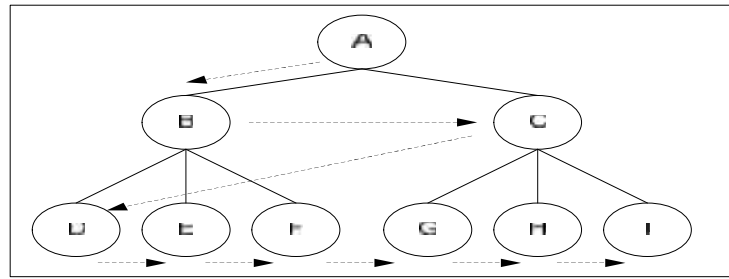
Pencarian adalah suatu tindakan untuk menentukan sesuatu. Hal terpenting dalam menentukan keberhasilan sistem yang berlandaskan AI adalah kesuksesan dalam pencarian dan pencocokan. Pada dasarnya ada 2 teknik pencarian yang digunakan, yaitu pencarian buta (*blind search*) dan pencarian terbimbing (*heuristic search*). (Suyoto, 2004)

2.4.1 Pencarian Buta (*Blind Search*)

2.4.1.1 Pencarian Melebar Pertama (*Breadth First Search*)

Breadth First Search adalah algoritma yang melakukan pencarian secara melebar yang mengunjungi suatu simpul kemudian mengunjungi semua simpul yang bertetangga dengan simpul tersebut terlebih dahulu (Erdi Wijaya, 2005). Algoritma ini memerlukan sebuah antrian *q* untuk menyimpan simpul yang telah dikunjungi. Simpul-simpul ini diperlukan sebagai acuan untuk mengunjungi simpul-simpul yang bertetangganya. Tiap simpul yang telah dikunjungi masuk ke dalam antrian hanya satu kali. Algoritma ini juga membutuhkan *table Boolean* untuk menyimpan simpul yang telah dikunjungi sehingga tidak ada simpul yang dikunjungi lebih dari satu kali.

Metode *Breadth-First Search*, semua *node* pada level *n* akan dikunjungi terlebih dahulu sebelum mengunjungi *node-node* pada level *n+1*. Pencarian dimulai dari *node* akar terus ke level ke-1 dari kiri ke kanan, kemudian berpindah ke level berikutnya. Demikian seterusnya hingga ditemukannya solusi.



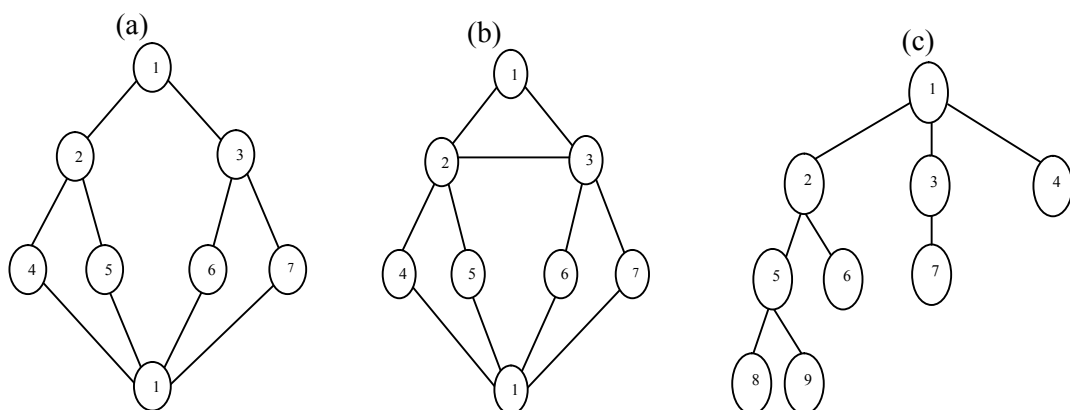
Gambar 2.3 Metode *Breadth First Search*

2.4.1.1.1 Cara Kerja Algoritma BFS

Dalam algoritma BFS, simpul anak yang telah dikunjungi disimpan dalam suatu antrian. Antrian ini digunakan untuk mengacu simpul-simpul yang bertetangga dengannya yang akan dikunjungi kemudian sesuai urutan pengantrian. Untuk memperjelas cara kerja algoritma BFS beserta antrian yang menggunakannya, berikut langkah-langkah algoritma BFS:

1. Masukkan simpul ujung (akar) ke dalam antrian
2. Ambil simpul dari awal antrian, lalu cek apakah simpul merupakan solusi
3. Jika simpul merupakan solusi, pencarian selesai dan hasil dikembalikan.
4. Jika simpul bukan solusi, masukkan seluruh simpul yang bertetangga dengan simpul tersebut (simpul anak) ke dalam antrian
5. Jika antrian kosong dan setiap simpul sudah dicek, pencarian selesai dan mengembalikan hasil solusi tidak ditemukan
6. Ulangi pencarian dari langkah kedua

Contohnya terlihat dibawah ini:



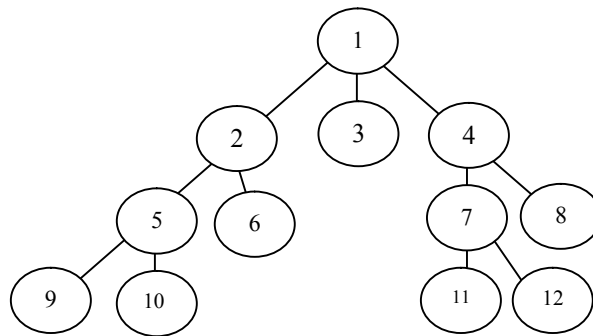
Gambar 2.4 Pohon Permasalahan

Maka penyelesaiannya adalah:

- Gambar (a) BFS(1): 1, 2, 3, 4, 5, 6, 7, 1.
- Gambar (b) BFS(1): 1, 2, 3, 4, 5, 6, 7, 1
- Gambar (c) BFS(1): 1, 2, 3, 4, 5, 6, 7, 8, 9

Adapun contoh untuk mencari lintasan terpendek dengan menggunakan algoritma BFS adalah sebagai berikut:

Diketahui sebuah kota, dengan memiliki inisial seperti yang ditunjukkan dibawah ini. Jarak antar kota dibentuk dengan sebuah graph terlihat dibawah:



Gambar 2.5 Pohon Rute Permasalahan

Pertanyaan: sebutkan rute yang akan ditempuh untuk mencapai kota no. 8. Titik awal perjalanan adalah kota no. 1. Gunakan algoritma BFS!

Maka dengan menggunakan algoritma BFS, rute tercepat yang didapat adalah sebagai berikut: 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8. Rute tersebut didapat dari pencarian secara melebar. Hal tersebut dapat dijabarkan sebagai berikut:

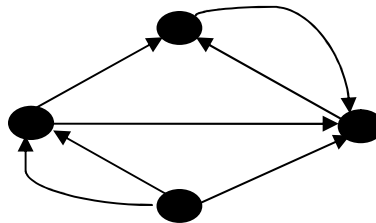
1. Pertama, pointer menunjuk pada daun yang ada sebelah kanan, yaitu no.2 (1 – 2)
2. Setelah itu, proses dilanjutkan pada tetangga no.2 yaitu no.3 (1-2-3) dan selanjutnya mengarah pada tetangga terdekat, yakni no.4 (1-2-3-4).
3. Pointer mencari tetangga no.4, namun karna tidak ada, maka pointer kembali ke kota no.2 dan masuk ke daun berikutnya, yakni no.5.
4. Proses diulang hingga pointer menunjuk angka 8

2.4.1.1.2 Aplikasi Algoritma BFS Dalam Teori Graf

Berdasarkan orientasi pada arah sisi, maka secara umum graf dibedakan atas 2 jenis :

1. Graf berarah (*Directed Graph*)

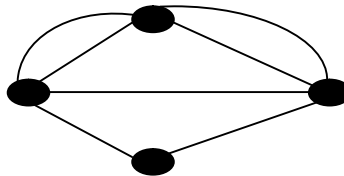
Yaitu graf yang setiap sisinya diberikan orientasi arah.



Gambar 2.6 graf berarah

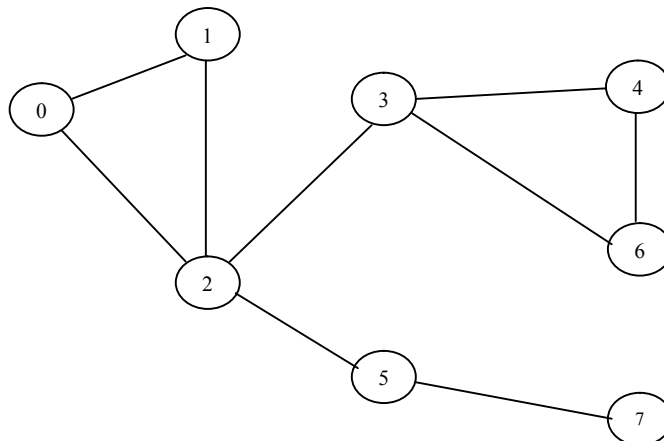
2. Graf tidak berarah (*Undirected Graph*)

Adalah graf yang tidak memiliki orientasi arah pada setiap sisinya.



Gambar 2.7 graf tidak berarah

Penelusuran simpul-simpul pada graf tidak berarah (*undirected graph*) dengan metode BFS dilakukan dengan cara mengunjungi tetangga terdekat dari simpul awal/simpul yang telah ditentukan. Sebagai contoh :



Gambar 2.8 contoh graf tidak berarah (*undirected graph*)

Berdasarkan gambar 2.8 diatas, jika diselesaikan dengan menggunakan metode BFS maka pengunjungan simpul-simpul akan dilakukan sebagai berikut :

1. Simpul awal 0, urutan simpul yang dikunjungi adalah 0,1,2,3,5,4,6,7
2. Simpul awal 1, urutan simpul yang dikunjungi adalah 1,0,2,3,5,4,6,7
3. Simpul awal 2, urutan simpul yang dikunjungi adalah 2,0,1,3,5,4,6,7
4. Simpul awal 3, urutan simpul yang dikunjungi adalah 3,2,4,6,0,1,5,7
5. Simpul awal 4, urutan simpul yang dikunjungi adalah 4,3,6,2,0,1,5,7
6. Simpul awal 5, urutan simpul yang dikunjungi adalah 5,2,7,0,1,3,4,6
7. Simpul awal 6, urutan simpul yang dikunjungi adalah 6,3,4,2,0,1,5,7
8. Simpul awal 7, urutan simpul yang dikunjungi adalah 7,5,2,0,1,3,4,6

Keuntungan dari metode BFS yaitu :

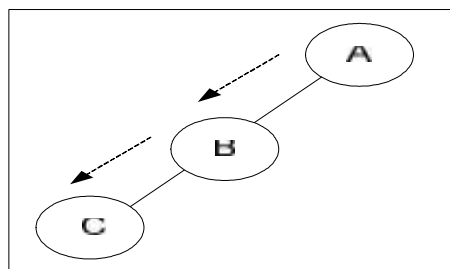
1. Tidak akan menemui jalan buntu.
2. Jika ada satu solusi, maka *Breadth First Search* akan menemukannya. Dan jika ada lebih dari satu solusi, maka solusi minimum akan ditemukan.

Kelemahan dari metode BFS yaitu :

1. Membutuhkan memori yang cukup banyak, karena menyimpan semua *node* dalam satu pohon.
2. Membutuhkan waktu yang cukup lama, karena akan menguji n level untuk mendapatkan solusi pada level ke- $(n + 1)$. (Kusumadewi.S, 2002)

2.4.1.2 Pencarian Mendalam Pertama (Depth-First Search)

Pada metode *Depth-First Search*, proses pencarian akan dilakukan pada semua anaknya sebelum dilakukan pencarian ke *node-node* yang selevel. Pencarian dimulai dari *node* akar ke level yang lebih tinggi. Proses ini diulangi terus hingga ditemukannya solusi.



Gambar 2.9 Metode *Depth-First Search*

Keuntungan dari metode ini:

1. Membutuhkan memori yang relatif kecil, karena hanya *node-node* pada lintasan yang aktif saja yang disimpan.
2. Secara kebetulan, metode *Depth First Search* akan menemukan solusi tanpa harus menguji lebih banyak lagi dalam ruang keadaan.

Kelemahan dari metode ini:

1. Memungkinkan tidak ditemukannya tujuan yang diharapkan.
2. Hanya akan mendapatkan 1 solusi pada setiap pencarian. (Kusumadewi.S, 2002)

2.5 Asal Usul Permainan Catur

2.5.1 Catur

Catur adalah permainan mental yang dimainkan oleh dua orang. Pecatur adalah orang yang memainkan catur, baik dalam pertandingan satu lawan satu maupun satu melawan banyak orang (dalam keadaan informal). Sebelum bertanding, pecatur memilih biji catur yang akan ia mainkan. Terdapat dua warna yang membedakan bidak atau biji catur, yaitu hitam dan putih. Pemegang buah putih memulai langkah pertama, yang selanjutnya diikuti oleh pemegang buah hitam secara bergantian sampai permainan selesai. Kata catur diambil dari bahasa Sanskerta yang berarti "empat". Namun kata ini sebenarnya merupakan singkatan dari *caturangga* yang berarti empat sudut. Di India kuno permainan catur memang dimainkan oleh empat peserta yang berada di empat sudut yang berbeda. Hal ini lain dari permainan catur modern di mana pesertanya hanya dua orang saja. Kemudian kata *caturangga* ini diserap dalam bahasa Persia menjadi *shatranj*. Kata *chess* dalam bahasa Inggris diambil dari bahasa Persia *shah*. (Bey Magethi. 2009)

2.5.2 Gerakan

Sebelum bertanding, pecatur memilih warna buah yang akan ia mainkan. Pemegang buah putih memulai langkah pertama, yang selanjutnya diikuti oleh pemegang buah hitam secara bergantian. Setiap langkah hanya boleh menggerakkan satu bidak saja (kecuali untuk rokade dimana ada dua bidak yang

digerakkan). Bidak dipindahkan ke petak kosong, ataupun yang ditempati oleh bidak lawan, yang berarti menangkapnya dan memindahkan bidak lawan dari permainan. Setiap bidak catur memiliki gerakan yang unik sebagai berikut:

- Raja, dapat bergerak satu petak ke segala arah. Raja juga memiliki gerakan khusus yang disebut rokade yang turut melibatkan sebuah benteng.
- Benteng, dapat bergerak sepanjang petak horizontal maupun vertikal, tetapi tidak dapat melompati bidak lain. Seperti yang telah di atas, benteng terlibat dalam gerakan rokade.
- Gajah/Rencong, dapat bergerak sepanjang petak secara diagonal, tetapi tidak dapat melompati bidak lain.
- Ratu, memiliki gerakan kombinasi dari Benteng dan Gajah.
- Kuda, memiliki gerakan mirip huruf L, yaitu memanjang dua petak dan melebar satu petak. Kudalah satu-satunya bidak yang dapat melompati bidak-bidak lain. (<http://id.wikipedia.org>)

Bidak/Pion, dapat bergerak maju (arah lawan) satu petak ke petak yang tidak ditempati. Pada gerakan awal, pion dapat bergerak maju dua petak. Pion juga dapat menangkap bidak lawan secara diagonal, apabila bidak lawan tersebut berada satu petak di diagonal depannya. Pion memiliki dua gerakah khusus, yaitu gerakan menangkap *en passant* dan promosi. *En passant* dapat terjadi ketika pion bergerak dua petak maju dan ada pion lawan yang berada satu petak dalam baris tujuan, maka pion lawan dapat menangkap dan menempati petak yang baru saja dilalui pion tersebut (seolah-olah pion tersebut bergerak satu petak maju). Namun demikian, gerakan ini hanya dapat dilakukan sesaat setelah gerakan pion maju dua petak, atau hak lawan untuk melakukan gerakan *en passant* ini hilang. Sedangkan promosi akan terjadi ketika pion telah maju hingga menempati baris paling akhir, berbarengan dengan gerakan maju tersebut, pion dipromosikan dan harus ditukar dengan bidak berdasarkan keinginan pemain, yaitu Ratu, Benteng, Gajah, ataupun Kuda dengan warna yang sama. Pada umumnya, pion dipromosikan menjadi Ratu. Tidak ada peraturan yang membatasi bidak yang dipilih sebagai promosi, jadi dimungkinkan memiliki bidak yang melebihi jumlahnya waktu awal permainan (semisal, dua Ratu). (<http://id.wikipedia.org>)

2.5.3 Rokade

Rokade (dalam bahasa Inggris, *castling*) merupakan gerakan khusus dalam catur di mana Raja bergerak dua petak menuju Benteng di baris pertamanya, kemudian meletakkan Benteng pada petak terakhir yang dilalui Raja. Persyaratan rokade adalah sebagai berikut:

- Bidak Raja dan Benteng yang akan dilibatkan dalam rokade harus belum pernah bergerak
- Tidak ada bidak lain di antara Raja dan Benteng
- Raja tidak sedang di-skak, dan petak-petak yang dilalui Raja tidak sedang *diserang* oleh bidak lawan

Hal-hal berikut ini merupakan kesalah pengertian dalam rokade, yang semestinya tidak berlaku:

- Bidak benteng yang terlibat rokade sedang diserang

Jika benteng yang dilibatkan berada di sisi Ratu, petak yang berada persis di samping Benteng tersebut tidak boleh dalam serangan. (<http://id.wikipedia.org>).

2.5.4 Skak

Ketika Raja sedang diserang oleh satu atau lebih bidak lawan, keadaan ini disebut dengan skak. Pemain yang Rajanya diskak harus menggerakkan Rajanya supaya tidak terserang. Hal ini dapat dilakukan dengan menangkap bidak lawan yang menyerang, menutup serangan lawan dengan menempatkan sebuah bidak di antaranya (apabila yang menyerang Ratu, Benteng, atau Gajah dan ada petak kosong di antara Raja dan bidak lawan), atau memindahkan Raja ke petak yang tidak sedang diserang. Rokade tidak diijinkan apabila Raja sedang diskak. (<http://id.wikipedia.org>)

2.5.5 Akhir Permainan

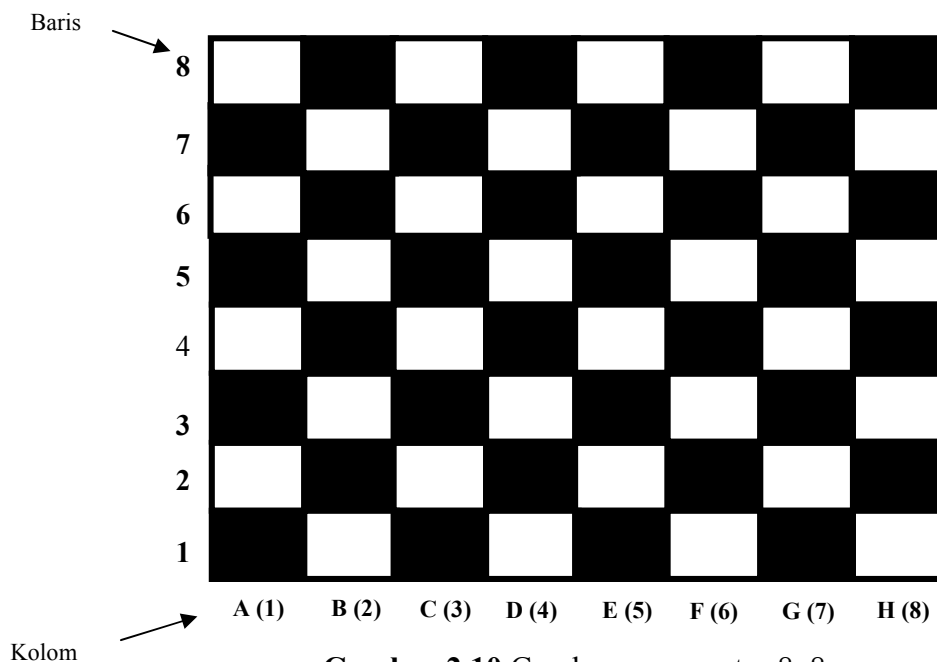
Tujuan permainan adalah mencapai posisi skak mat. Hal ini bisa terjadi bila Raja terancam dan tidak bisa menyelamatkan diri ke petak lain. Tidak selalu permainan berakhir dengan kekalahan, karena bisa terjadi pula peristiwa seri atau remis di mana kedua belah pihak tidak mampu lagi meneruskan pertandingan karena tidak bisa mencapai skak mat. Peristiwa remis ini bisa terjadi berdasarkan kesepakatan maupun tidak. Salah satu contoh remis yang tidak berdasarkan

kesepakatan tetapi terjadi adalah pada keadaan remis abadi. Keadaan remis yang lain adalah keadaan pat, dimana yang giliran melangkah tidak bisa melangkahkan buah apapun termasuk Raja, tetapi tidak dalam keadaan terancam skak. Dalam pertandingan catur pihak yang menang biasanya mendapatkan nilai 1, yang kalah 0, sedang draw 0.5. Indonesia saat ini sudah mempunyai 7 pecatur yang bergelar Grandmaster (GM) yaitu :(<http://id.wikipedia.org>)

1. Utut Adianto
2. Edhi Handoko (Almarhum)
3. Cerdas Barus
4. Ruben Gunawan (Almarhum)
5. Ardiansyah
6. Susanto Megaranto (Grandmaster termuda)
7. Herman Suradiradja (Grandmaster pertama)

2.6 Permasalahan Langkah Kuda dalam Papan Catur

Permasalahan ini merupakan salah satu masalah klasik dalam *artificial intelligence* (AI). Sebuah biji kuda dalam papan catur memiliki pergerakan menyerupai huruf L. Biji catur ini merupakan salah satu biji yang paling sulit digerakkan dan sering juga merupakan biji yang paling berbahaya apabila tidak diperhatikan secara seksama setiap pergerakannya. Permasalahan ini menyediakan sebuah papan catur berukuran $n \times n$. Sasaran (*goal*) dari permasalahan ini adalah menggerakkan sebuah biji kuda dari suatu posisi tertentu pada papan catur ke posisi tujuan yang diinginkan dengan mensimulasikan semua pergerakan terpendek menuju ke posisi tujuan tanpa melanggar aturan-aturan yang telah ditetapkan. Penyelesaian permasalahan ini dapat menggunakan bantuan pohon pelacakan. (Sandi.S, 1993)



Gambar 2.10 Gambar papan catur 8x8

Dengan memperhatikan gambar 2.10 maka aturan-aturan yang terdapat pada permasalahan langkah kuda dalam papan catur adalah:

1. Kuda dapat digeser ke atas sebelah kiri.

Operasi: Baris = baris + 2, Kolom = kolom - 1.

Aturan: Nilai variabel kolom setelah operasi harus > 0 , nilai variabel baris ≤ 8 , dan tidak terdapat biji catur lain pada posisi tersebut.

2. Kuda dapat digeser ke atas sebelah kanan.

Operasi: Baris = baris + 2, Kolom = kolom + 1.

Aturan: Nilai variabel kolom dan baris setelah operasi harus ≤ 8 dan tidak terdapat biji catur lain pada posisi tersebut.

3. Kuda dapat digeser ke kiri sebelah atas.

Operasi: Baris = baris + 1, Kolom = kolom - 2.

4. Aturan: Nilai variabel kolom setelah operasi harus > 0 , nilai variabel baris ≤ 8 , dan tidak terdapat biji catur lain pada posisi tersebut. Kuda dapat digeser ke kiri sebelah bawah.

Operasi: Baris = baris - 1, Kolom = kolom - 2.

Aturan: Nilai variabel kolom dan baris setelah operasi harus > 0 dan tidak terdapat biji catur lain pada posisi tersebut.

5. Kuda dapat digeser ke kanan sebelah atas.

Operasi: Baris = baris + 1, Kolom = kolom + 2.

Aturan: Nilai variabel kolom dan baris setelah operasi harus ≤ 8 dan tidak terdapat biji catur lain pada posisi tersebut.

6. Kuda dapat digeser ke kanan sebelah bawah.

Operasi: Baris = baris – 1, Kolom = kolom + 2.

Aturan: Nilai variabel kolom setelah operasi harus ≤ 8 , nilai variabel baris harus > 0 , dan tidak terdapat biji catur lain pada posisi tersebut.

7. Kuda dapat digeser ke bawah sebelah kiri.

Operasi: Baris = baris – 2, Kolom = kolom – 1.

Aturan: Nilai variabel kolom dan baris setelah operasi harus > 0 dan tidak terdapat biji catur lain pada posisi tersebut.

8. Kuda dapat digeser ke bawah sebelah kanan.

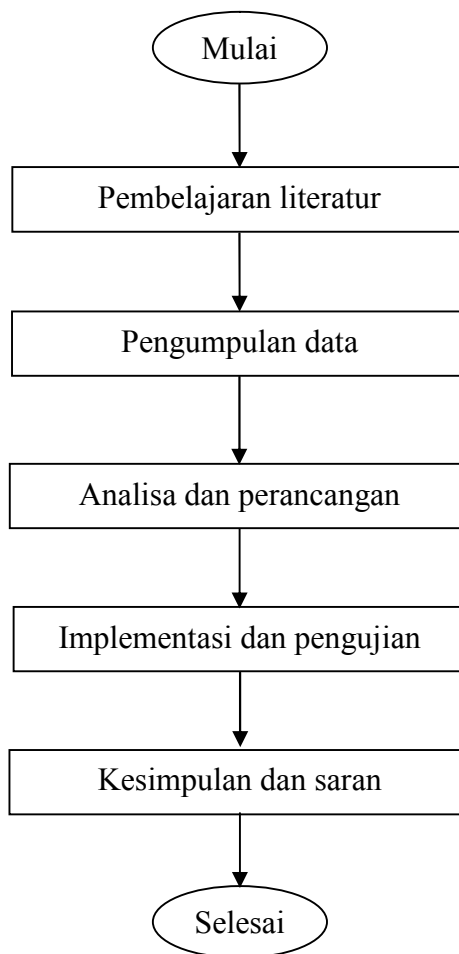
Operasi: Baris = baris – 2, Kolom = kolom + 1.

Aturan: Nilai variabel kolom setelah operasi harus ≤ 8 , nilai variabel baris harus > 0 , dan tidak terdapat biji catur lain pada posisi tersebut.

BAB III

METODOLOGI PENELITIAN

Dalam penelitian tugas akhir ini, metodologi penelitian merupakan pedoman dalam pelaksanaan penelitian sehingga yang dicapai tidak menyimpang dari tujuan yang telah ditentukan sebelumnya. Berikut gambaran langkah-langkah secara garis besar dalam penelitian ini :



Gambar 3.1 Diagram langkah-langkah penelitian

3.1 Pembelajaran Secara Literatur

Pada tahap ini dilakukan proses pengumpulan informasi yang diperlukan untuk proses perancangan aplikasi. Studi literatur ini mencakup tentang

pemahaman konsep. Konsep-konsep tersebut antara lain: mengenai kecerdasan buatan (AI), pemahaman metode BFS (*Breadth First Search*), serta pembelajaran mengenai Sistem Produksi.

3.2 Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data yang dibutuhkan dalam pembuatan aplikasi. Pengumpulan data ini dapat diperoleh dari artikel-artikel, jurnal, buku dan juga dari internet yang berhubungan untuk perancangan aplikasi ini.

3.3 Analisa Dan Perancangan

Analisa dilakukan setelah data yang dikumpulkan telah lengkap agar selanjutnya mulai merancang aplikasi. Pada saat menganalisa data, ada beberapa tahap yang harus dilakukan, yaitu mengidentifikasi kebutuhan aplikasi, fungsi aplikasi, merancang lingkungan implementasi, serta merancang antarmuka pengguna (*interface*) yang akan dibangun.

1. Perancangan *Interface*.

Merancang antarmuka untuk proses-proses utama yang ada dalam aplikasi

2. Perancangan *identification process*

Menggambarakan rancangan menu proses *identification image input*

Data yang digunakan dalam pembuatan aplikasi ini adalah *image* Kuda, Raja dan Bidak/Pion. Data tersebut dibagi menjadi data masukan, data proses, dan data keluaran. Data-data tersebut adalah:

2.1 Data masukan

Kuda diletakkan pada posisi yang diinginkan (sebagai posisi awal). Selanjutnya Raja diletakkan pada posisi yang diinginkan (posisi tujuan) dan Bidak/pion diletakkan pada posisi yang diinginkan pula sebagai rintangan yang tidak boleh ditempati oleh kuda (maksimal 8 bidak/pion)

2.2 Data Proses

Selanjutnya dilakukan proses pencarian langkah terpendek yang dapat dilakukan oleh kuda hingga mencapai posisi tujuan (Raja) dengan aturan –aturan yang telah ditetapkan dan tidak boleh menempati posisi

yang ditempati oleh bidak/pion dengan metode BFS (*Breadth First Search*).

2.3 data Keluaran

setelah dilakukan proses maka akan didapat data keluaran yaitu beberapa langkah terpendek yang dapat dilalui oleh kuda hingga mencapai posisi tujuan (Raja)

3.4 Implementasi Dan Pengujian

Tahap terakhir yaitu pembuatan aplikasi berdasarkan analisa yang telah dilakukan. Tahap implementasi merupakan tahap penulisan kode program berdasarkan analisa dan perancangan yang telah dilakukan. Bahasa pemrograman yang digunakan untuk membangun aplikasi ini adalah *Visual Basic 6.0*. Selanjutnya dilakukan pengujian.

3.5 Kesimpulan Dan Saran

Kesimpulan dan saran merupakan tahap akhir dari penelitian yang dilakukan. Dibagian ini akan ditarik kesimpulan berdasarkan hasil dari aplikasi yang telah dibuat serta memberikan saran-saran untuk menyempurnakan dan mengembangkan aplikasi ini.

BAB IV

ANALISA DAN PERANCANGAN

4.1 Analisa

Permasalahan pergerakan langkah kuda dalam papan catur merupakan salah satu masalah klasik dalam AI. Dalam permainan catur, biji kuda memiliki pergerakan menyerupai huruf L. Biji catur ini merupakan salah satu biji yang paling sulit digerakkan dan sering juga merupakan biji yang paling berbahaya apabila tidak diperhatikan secara seksama setiap pergerakannya. Permasalahan ini menyediakan sebuah papan catur berukuran 8 x 8. Sasaran (*goal*) dari permasalahan ini adalah menggerakkan sebuah biji kuda dari suatu posisi tertentu pada papan catur ke posisi tujuan yang diinginkan dengan mensimulasikan semua pergerakan yang mungkin untuk menuju ke posisi tujuan tanpa melanggar aturan-aturan yang telah ditetapkan.

Variabel-variabel yang terdapat pada perangkat lunak adalah:

1. Kuda (putih): merupakan simbol posisi awal.



Gambar 4.1 Simbol Posisi Awal

2. Raja (hitam): merupakan simbol posisi tujuan.



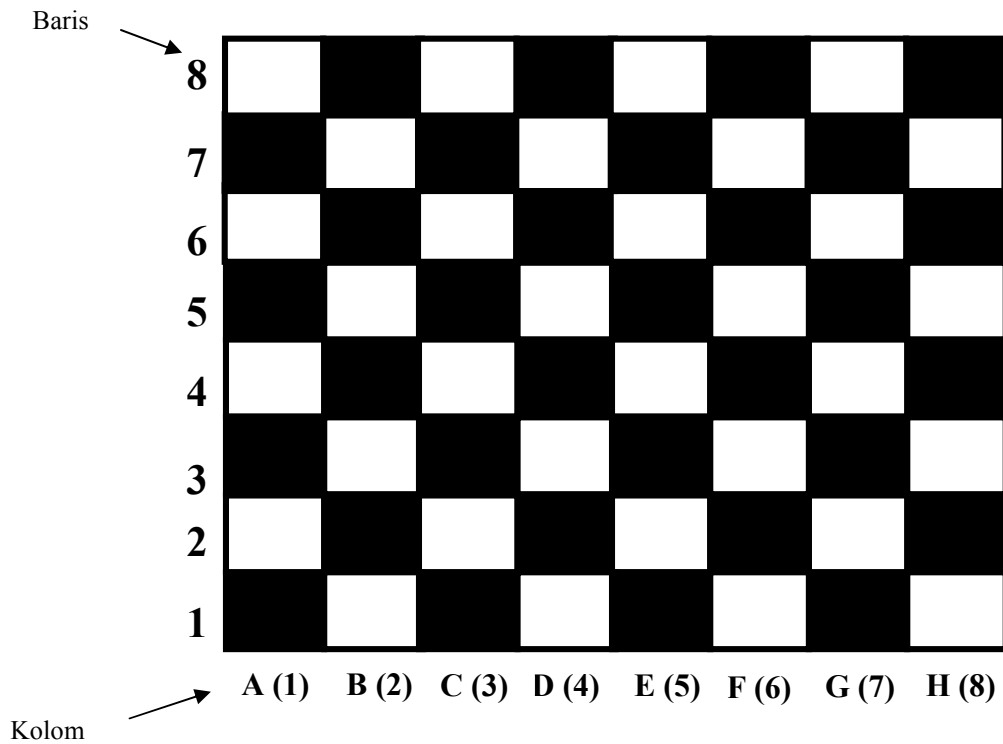
Gambar 4.2 Simbol Posisi Tujuan

3. Bidak/pion (hitam): merupakan penghalang yang tidak boleh ditempati oleh kuda .



Gambar 4.3 Simbol Rintangan

Wadah tempat pergerakan kuda adalah sebuah papan catur dengan ukuran 8 x 8 seperti terlihat pada gambar 4.4.



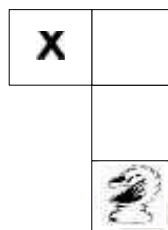
Gambar 4.4 Papan Catur 8 x 8

Permasalahan yang dihadapi adalah kuda harus digerakkan dari posisinya ke posisi raja dalam sebuah papan catur 8 x 8. Adapun aturan pergerakan biji kuda yang mungkin adalah:

1. Kuda dapat digeser ke atas sebelah kiri.

Operasi: Baris = baris + 2, Kolom = kolom – 1.

Aturan: Nilai variabel kolom setelah operasi harus > 0 , nilai variabel baris ≤ 8 , dan tidak terdapat biji bidak beserta langkah aktif bidak pada posisi tersebut

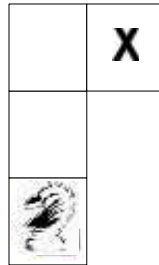


Gambar 4.5 Aturan Pergerakan-1

2. Kuda dapat digeser ke atas sebelah kanan.

Operasi: Baris = baris + 2, Kolom = kolom + 1.

Aturan: Nilai variabel kolom setelah operasi harus ≤ 8 dan baris setelah operasi harus ≤ 8 dan tidak terdapat biji bidak beserta langkah aktif bidak pada posisi tersebut.

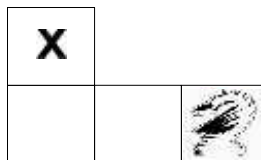


Gambar 4.6 Aturan Pergerakan-2

3. Kuda dapat digeser ke kiri sebelah atas.

Operasi: Baris = baris + 1, Kolom = kolom - 2.

Aturan: Nilai variabel kolom setelah operasi harus > 0 , nilai variabel baris ≤ 8 , dan tidak terdapat biji bidak beserta langkah aktif bidak pada posisi tersebut.

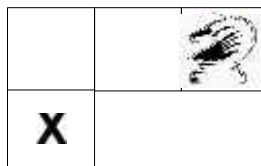


Gambar 4.7 Aturan Pergerakan-3

4. Kuda dapat digeser ke kiri sebelah bawah.

Operasi: Baris = baris - 1, Kolom = kolom - 2.

Aturan: Nilai variabel kolom dan baris setelah operasi harus > 0 dan tidak terdapat biji bidak beserta langkah aktif bidak pada posisi tersebut.

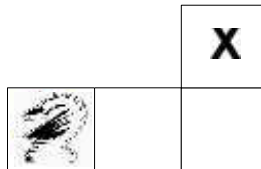


Gambar 4.8 Aturan Pergerakan-4

5. Kuda dapat digeser ke kanan sebelah atas.

Operasi: Baris = baris + 1, Kolom = kolom + 2.

Aturan: Nilai variabel kolom setelah operasi harus ≤ 8 dan baris setelah operasi harus ≤ 8 dan tidak terdapat biji bidak beserta langkah aktif bidak pada posisi tersebut.

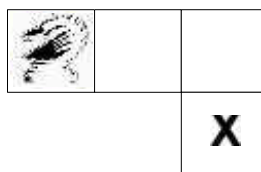


Gambar 4.9 Aturan Pergerakan-5

6. Kuda dapat digeser ke kanan sebelah bawah.

Operasi: Baris = baris - 1, Kolom = kolom + 2.

Aturan: Nilai variabel kolom setelah operasi harus ≤ 8 , nilai variabel baris harus > 0 , dan tidak terdapat biji bidak beserta langkah aktif bidak pada posisi tersebut.

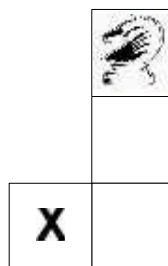


Gambar 4.10 Aturan Pergerakan-6

7. Kuda dapat digeser ke bawah sebelah kiri.

Operasi: Baris = baris - 2, Kolom = kolom - 1.

Aturan: Nilai variabel kolom dan baris setelah operasi harus > 0 dan tidak terdapat biji bidak beserta langkah aktif bidak pada posisi tersebut.

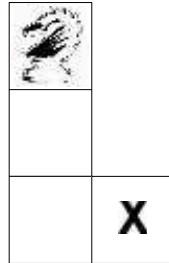


Gambar 4.11 Aturan Pergerakan-7

8. Kuda dapat digeser ke bawah sebelah kanan.

Operasi: Baris = baris - 2, Kolom = kolom + 1.

Aturan: Nilai variabel kolom setelah operasi harus ≤ 8 , nilai variabel baris harus > 0 , dan tidak terdapat biji bidak beserta langkah aktif bidak pada posisi tersebut.



Gambar 4.12 Aturan Pergerakan-8

Algoritma yang dibutuhkan dalam merancang Aplikasi Simulasi Langkah Kuda Dalam Papan Catur dapat dibagi menjadi 2 (dua) bagian, yaitu :

1. Algoritma pencarian solusi.
2. Algoritma fungsi pendukung.

4.1.1 Algoritma Pencarian Solusi

Seperti telah dijelaskan pada bab 2 bahwa pencarian solusi menggunakan metode *Breadth-First Search* (BFS). Algoritma pencarian BFS akan melakukan penelusuran *node* per level, artinya semua *node* pada level yang sama akan dikunjungi terlebih dahulu sebelum penelusuran dilanjutkan ke *node* pada level berikutnya. Karena bisa terdapat lebih dari satu solusi, maka penelusuran atau pengembangan *node* dilakukan hingga tidak ada *node* yang dapat dikembangkan lagi dan *node* yang berisi posisi tujuan merupakan solusi.

Terdapat 8 buah operasi untuk mengembangkan setiap *node* dalam pohon pencarian, yaitu:

1. Geser biji kuda ke atas sebelah kiri (Baris = baris + 2, Kolom = kolom - 1).
2. Geser biji kuda ke atas sebelah kanan (Baris = baris + 2, Kolom = kolom + 1).
3. Geser biji kuda ke kiri sebelah atas (Baris = baris + 1, Kolom = kolom - 2).
4. Geser biji kuda ke kiri sebelah bawah (Baris = baris - 1, Kolom = kolom - 2).
5. Geser kuda ke kanan sebelah atas (Baris = baris + 1, Kolom = kolom + 2).
6. Geser kuda ke kanan sebelah bawah (Baris = baris - 1, Kolom = kolom + 2).
7. Geser kuda ke bawah sebelah kiri (Baris = baris - 2, Kolom = kolom - 1).
8. Geser kuda ke bawah sebelah kanan (Baris = baris - 2, Kolom = kolom + 1).

Operasi ini harus mematuhi aturan sebagai berikut:

1. Posisi setelah pergeseran tidak boleh berada di luar papan catur.
2. Posisi setelah pergeseran tidak boleh sama dengan posisi biji bidak/pion (rintangan)
3. Posisi setelah pergeseran belum pernah dilintasi sebelumnya (pada satu jalur yang sama dalam pohon pencarian). Ini menjamin bahwa biji kuda tidak akan menempati posisi yang sudah pernah ditempati sebelumnya.

4.1.2 Algoritma Fungsi Pendukung

Beberapa algoritma fungsi pendukung pada aplikasi ini adalah:

1. Fungsi ‘ConvIndexToPosisi’.

Fungsi ini mengkonversi nilai indeks kotak menjadi posisi pada papan catur.

Untuk lebih memahami cara kerja fungsi, perhatikan gambar 4.13 berikut.

BARIS 8	Index = 57	Index = 58	Index = 59	Index = 60	Index = 61	Index = 62	Index = 63	Index = 64
7	Index = 49	Index = 50	Index = 51	Index = 52	Index = 53	Index = 54	Index = 55	Index = 56
6	Index = 41	Index = 42	Index = 43	Index = 44	Index = 45	Index = 46	Index = 47	Index = 48
5	Index = 33	Index = 34	Index = 35	Index = 36	Index = 37	Index = 38	Index = 39	Index = 40
4	Index = 25	Index = 26	Index = 27	Index = 28	Index = 29	Index = 30	Index = 31	Index = 32
3	Index = 17	Index = 18	Index = 19	Index = 20	Index = 21	Index = 22	Index = 23	Index = 24
2	Index = 9	Index = 10	Index = 11	Index = 12	Index = 13	Index = 14	Index = 15	Index = 16
1	Index = 1	Index = 2	Index = 3	Index = 4	Index = 5	Index = 6	Index = 7	Index = 8
KOLOM	A(1)	B(2)	C(3)	D(4)	E(5)	F(6)	G(7)	H(8)

Gambar 4.13 Posisi dalam papan catur

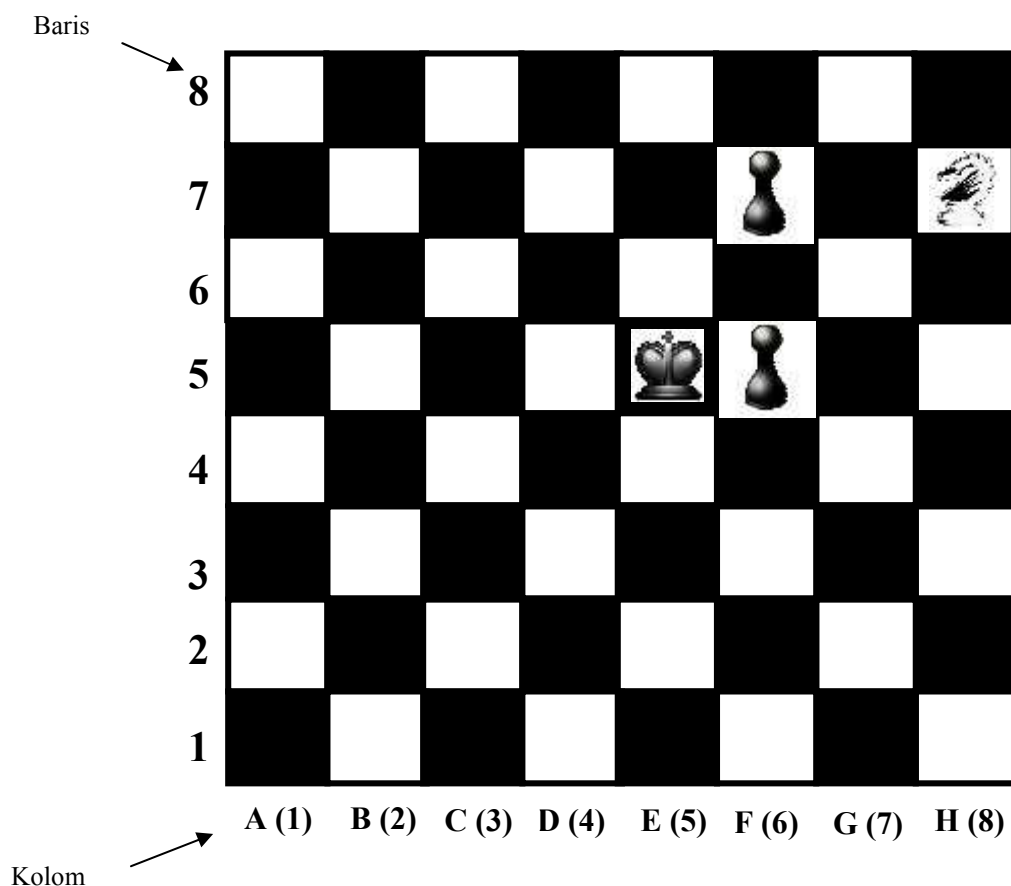
Tata cara penomoran posisi papan catur dalam aplikasi seperti terlihat pada gambar 4.13 Apabila $\text{index} = 30$, maka fungsi `ConvIndexToPosisi` akan mengembalikan nilai variabel `.Baris = 4`, `.Kolom = 6` dan `.Posisi = "F4"`.

2. Fungsi 'IsAllNodeCek'.

Fungsi ini melakukan pemeriksaan apakah semua *node* sudah ditelusuri. Fungsi mengembalikan nilai *boolean*. 'True' jika semua *node* sudah ditelusuri dan 'False' jika terdapat *node* yang belum ditelusuri.

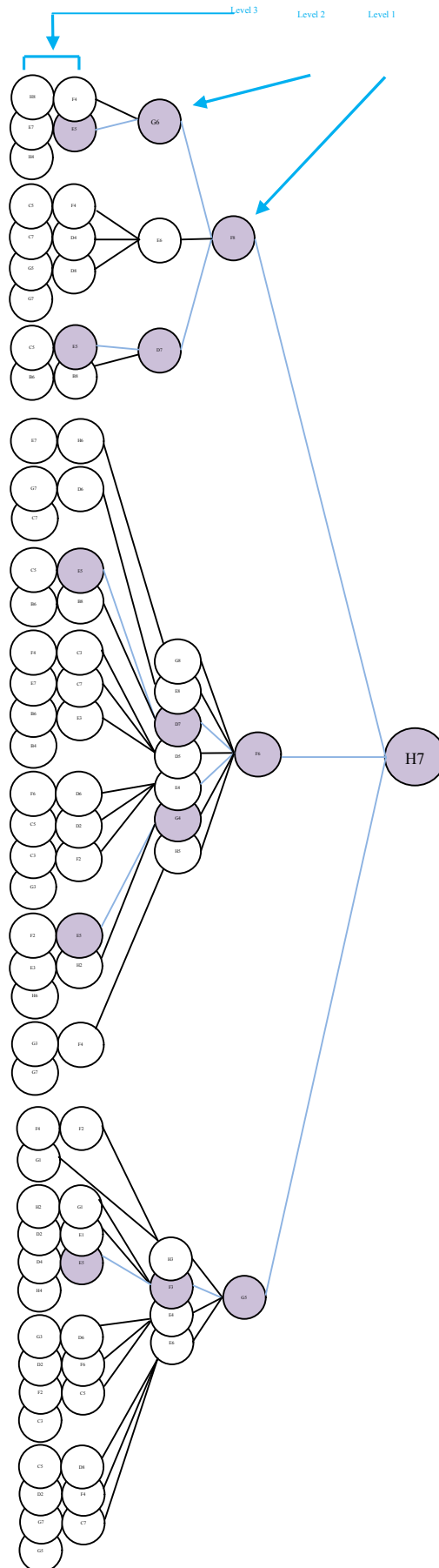
3. Fungsi 'IsValidPosisi'.

Fungsi melakukan pemeriksaan apakah suatu posisi *valid* atau tidak. Fungsi mengembalikan nilai *boolean*. 'True' jika posisi *valid* dan 'False' jika posisi tidak *valid*.



Gambar 4.14 Contoh permasalahan kuda

Pencarian solusi problema kuda dengan menggunakan metode BFS seperti terlihat pada gambar 4.15 berikut.



Gambar 4.15 Pencarian BFS pada problema kuda

Solusi terpendek dari permasalahan pada gambar 4.15 dapat dilihat pada tabel berikut.

Tabel 4.1 Solusi Permasalahan Kuda

No.	Solusi	Langkah-Langkah	Panjang
1.	Solusi-1	F8, D7, E5	3 langkah
2.	Solusi-2	F8, G6, E5	3 langkah
3	Solusi-3	F6, D7, E5	3 langkah
4	Solusi-4	F6, G4, E5	3 langkah
5	Solusi-5	G5, F3, E5	3 langkah

4.2 Perancangan

Perancangan aplikasi menggunakan bahasa pemrograman *Microsoft Visual Basic 6.0* dengan menggunakan beberapa komponen *standard visual basic*, seperti:

1. *Command Button*, sebagai tombol.
2. *Text Box*, sebagai tempat *input*.
3. *Image & PictureBox*, untuk menampilkan gambar.
4. *MSFlexGrid*, sebagai tabel dan komponen lainnya.

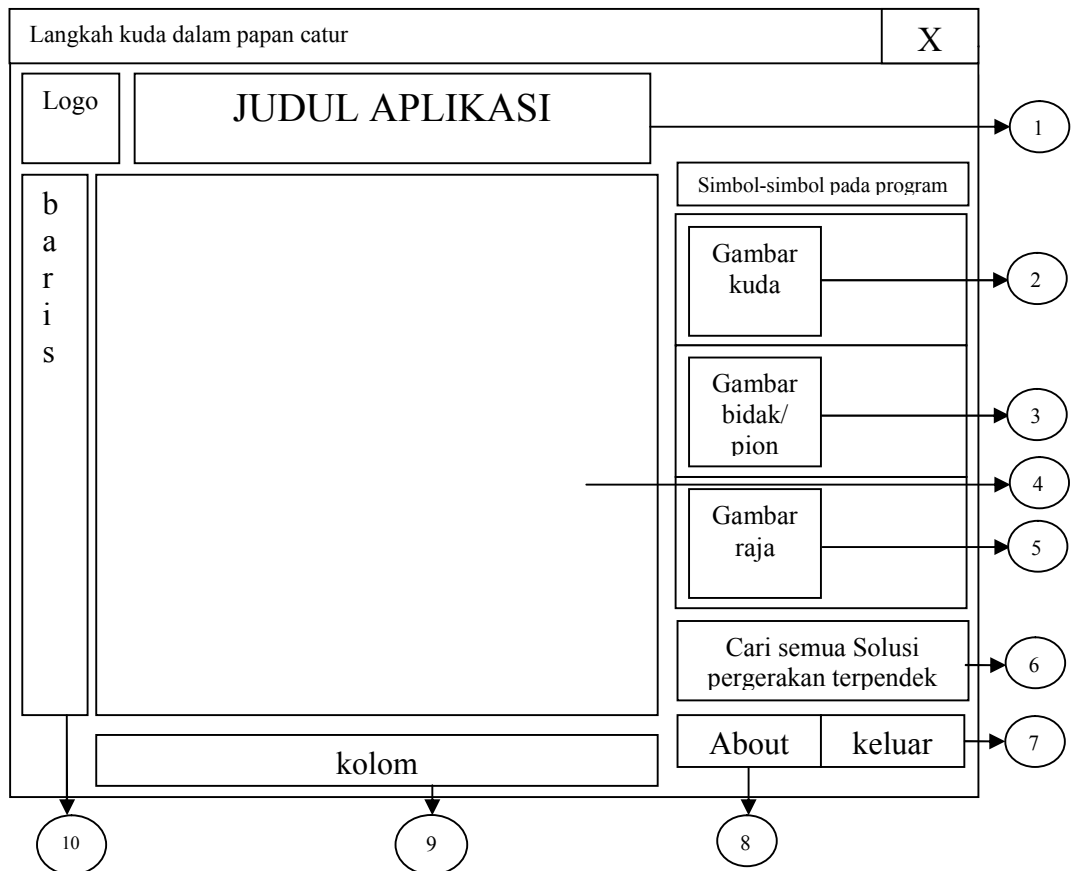
Aplikasi memiliki 3 (tiga) buah *form*, antara lain:

1. *Form Input*.
2. *Form Solusi*.
3. *Form About*.

Penjelasan perancangan dan fungsi masing-masing *form* akan disertakan pada subbab-subbab berikutnya.

4.2.1 Form Input

Form Input berfungsi untuk mengatur posisi awal, posisi tujuan dan posisi rintangan dalam permasalahan kuda.



Gambar 4.16 Rancangan *Form Input*

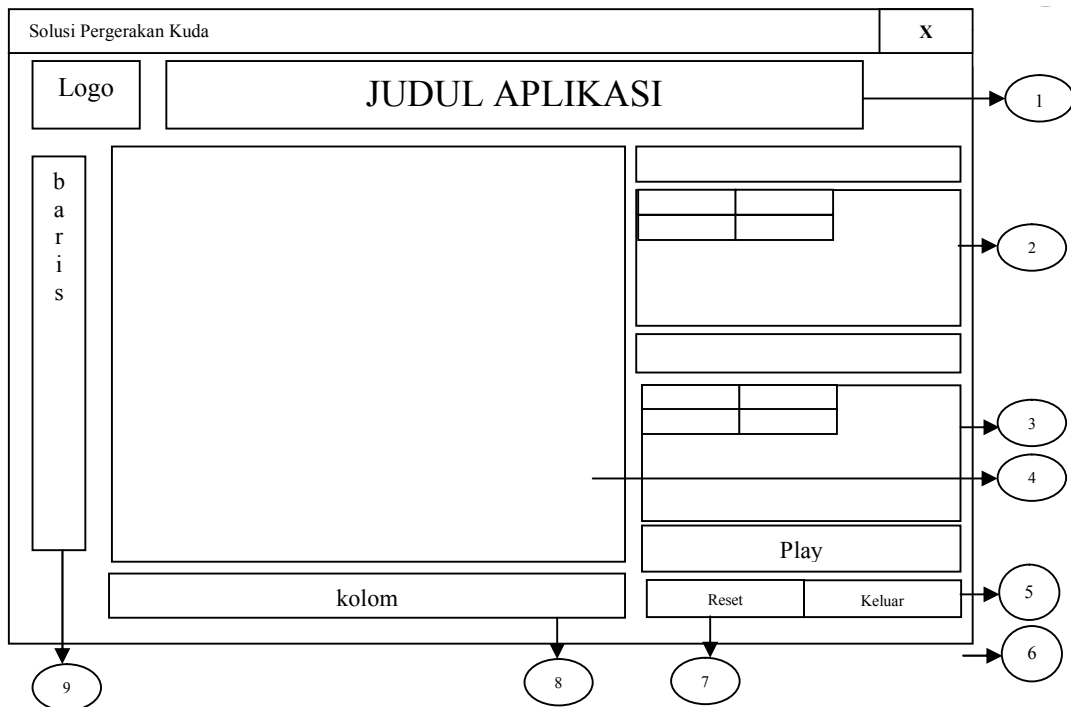
Keterangan:

Tabel 4.2 Rancangan *Form Input*

No	Nama objek	Keterangan
1	<i>TextBox</i>	Judul aplikasi
2	<i>Image</i> kuda	Posisi awal
3	<i>Image</i> bidak/pion	Posisi rintangan
4	<i>Image</i> raja	Posisi tujuan
5	<i>PictureBox</i>	Daerah papan catur
6	<i>CommandButton</i>	Tombol cari semua solusi pergerakan terpendek
7	<i>CommandButton</i>	Keluar dari aplikasi
8	<i>CommandButton</i>	Menampilkan <i>form About</i>
9	<i>TextBox</i>	Kolom
10	<i>TextBox</i>	Baris

4.2.2 Form Solusi

Form Solusi berfungsi untuk menampilkan semua solusi yang dihasilkan dan mensimulasikan gerakan kuda berdasarkan solusi yang dipilih.



Gambar 4.17 Rancangan *Form Solusi*

Keterangan:

Tabel 4.3 Rancangan *Form Solusi*

No	Nama objek	Keterangan
1	<i>TextBox</i>	Judul aplikasi
2	<i>MSFlexGrid</i>	Tabel untuk menampilkan semua solusi yang ada
3	<i>MSFlexGrid</i>	Tabel untuk menampilkan langkah-langkah pada solusi yang dipilih pada tabel no.2
4	<i>PictureBox</i>	Daerah papan catur
5	<i>CommandButton</i>	Tombol untuk memulai simulasi gerakan kuda.
6	<i>CommandButton</i>	Tombol keluar dari <i>form</i> simulasi
7	<i>CommandButton</i>	Tombol untuk mengulangi simulasi gerakan kuda dari awal.
8	<i>TextBox</i>	Kolom
9	<i>TextBox</i>	Baris

4.2.3 Form About

Form About berfungsi untuk menampilkan nama, nomor induk mahasiswa (NIM), jurusan dan nama kampus penyusun tugas akhir. *Form About* juga menyertakan sedikit informasi mengenai aplikasi tersebut.

The diagram shows a rectangular window titled "About" with a close button "X" in the top right corner. The layout includes the following elements:

- 1**: A text box labeled "Judul Aplikasi" (Application Title).
- 2**: A large text box labeled "Informasi mengenai aplikasi yang dibuat" (Information about the application created).
- 3**: A text box labeled "Nama Pembuat" (Creator Name).
- 4**: A text box labeled "Nim" (Student ID).
- 5**: A text box labeled "Jurusan" (Department).
- 6**: An "OK" button.

Arrows point from the numbered circles to their respective UI elements: 1 to the title box, 2 to the information box, 3 to the creator name box, 4 to the NIM box, 5 to the department box, and 6 to the OK button.

Gambar 4.18 Rancangan *Form About*

Keterangan:

Tabel 4.4 Rancangan *Form About*

No	Nama objek	Keterangan
1	<i>TextBox</i>	Nama aplikasi
2	<i>TextBox</i>	Informasi mengenai aplikasi
3	<i>TextBox</i>	Nama pembuat
4	<i>TextBox</i>	NIM
5	<i>CommandButton</i>	Tombol untuk menutup <i>form</i>
6	<i>TextBox</i>	Jurusan

BAB V

IMPLEMENTASI DAN PENGUJIAN

Implementasi dan pengujian merupakan tahap yang dilakukan setelah tahap analisa dan perancangan selesai dikerjakan.

5.1 Implementasi

Implementasi merupakan kelanjutan dari tahap perancangan aplikasi yang telah didesain. Implementasi juga merupakan tahap pembangunan aplikasi menggunakan perangkat keras dan perangkat lunak yang telah ditetapkan.

5.1.1 Lingkungan Implementasi

Lingkungan implementasi sistem ada 2 (dua) yaitu: lingkungan perangkat keras dan lingkungan perangkat lunak.

5.1.1.1 Perangkat Keras

Perangkat keras yang digunakan dalam Rancang Bangun Simulasi Langkah Kuda Dalam Papan Catur ini menggunakan komputer dengan spesifikasi :

1. *Processor* : Intel Core 2 Duo 1.73 GHz
2. *Memory* : 2 GB
3. *Harddisk* : 500 GB

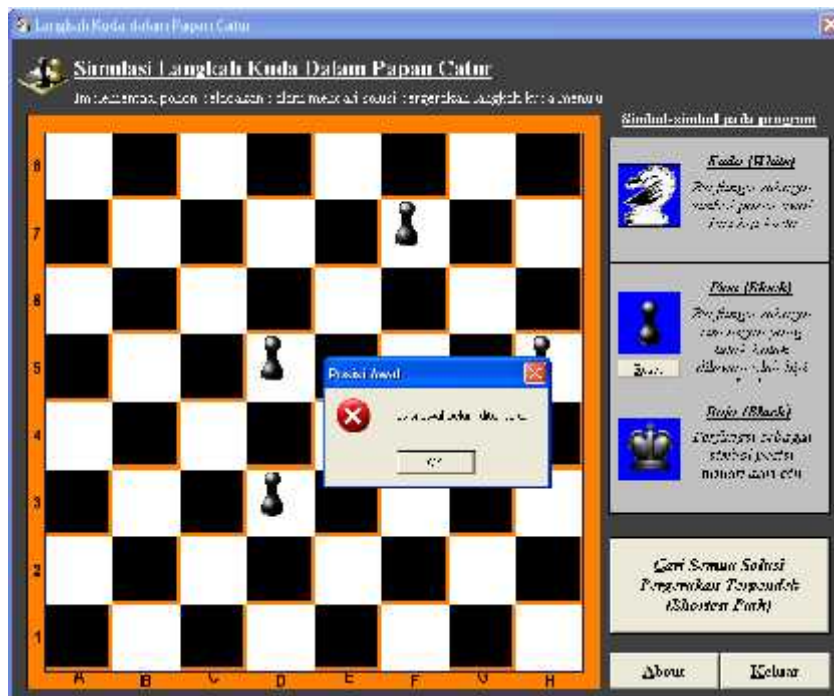
5.1.1.2 Perangkat lunak

Perangkat lunak yang digunakan dalam implelementasi ini menggunakan:

1. Sistem Operasi : *Windows XP*
2. Pemrograman : *Visual Basic 6.0*

5.1.2 Hasil Implementasi

Tampilan *form input error* pada saat posisi kuda (posisi awal) belum ditentukan terlihat pada gambar 5.1 berikut.



Gambar 5.1 Form input peringatan untuk posisi awal

Tampilan *form input error* pada saat posisi tujuan (posisi raja) belum ditentukan terlihat pada gambar 5.2 berikut.



Gambar 5.2 Form input peringatan untuk posisi tujuan

Sebagai contoh, *input* data adalah sebagai berikut.

➤ Contoh Kasus 1

1. Ukuran papan catur = 8 x 8.
2. Posisi biji kuda = H6
3. Posisi biji raja = C2
4. Posisi biji bidak = E6, F5, F7 dan H4

Tampilan *form input* untuk contoh kasus seperti terlihat pada gambar 5.3 berikut.



Gambar 5.3 Form Input (contoh kasus 1)



Gambar 5.4 Form Solusi (contoh kasus 1)

Solusi yang didapatkan seperti terlihat pada gambar 5.1 sebagai berikut:

Tabel 5.1 Solusi yang didapatkan (contoh kasus 1)

No	Solusi	Langkah-Langkah	Panjang Solusi
1	Solusi-1	G8, E7, C6, B4, C2	5 langkah
2	Solusi-2	G8, E7, C6, B4, C2	5 langkah
3	Solusi-3	G8, E7, D5, B4, C2	5 langkah
4	Solusi-4	G8, E7, D5, E3, C2	5 langkah
5	Solusi-5	G8, F6, D5, B4, C2	5 langkah
6	Solusi-6	G8, F6, D5, E3, C2	5 langkah

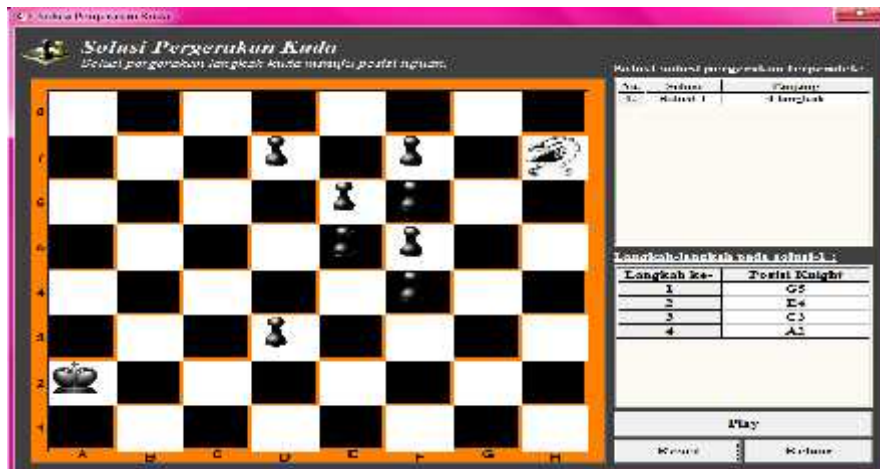
➤ Contoh kasus 2

1. Ukuran papan catur = 8 x 8.
2. Posisi biji kuda = H7
3. Posisi biji raja = A2
4. Posisi biji bidak = F4, F5, F6, F7, E5, E6, D3 dan D7

Tampilan *form input* untuk contoh kasus seperti terlihat pada gambar 5.5 berikut.



Gambar 5.5 *Form Input* (contoh kasus 2)



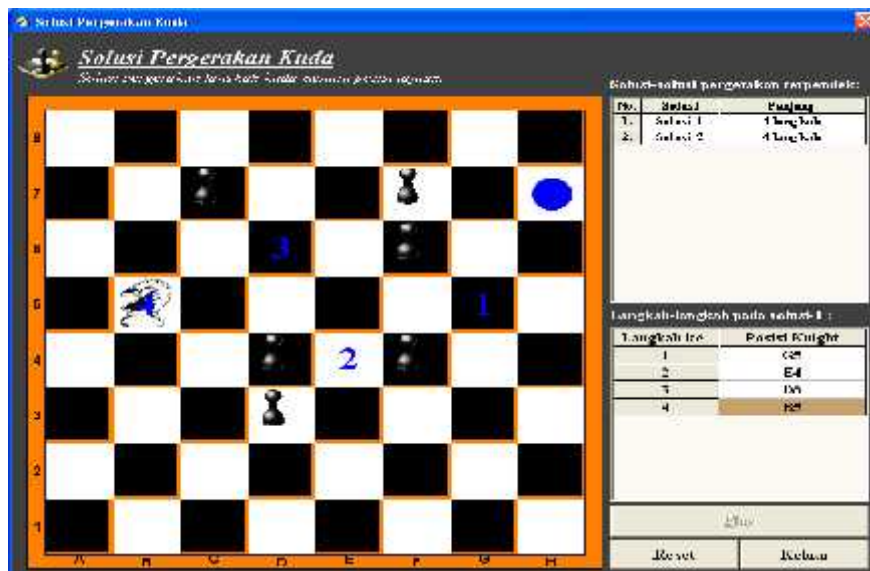
Gambar 5.6 Form Solusi (contoh kasus 2)

Solusi yang didapatkan seperti terlihat pada gambar 5.4 sebagai berikut:

Tabel 5.2 Solusi yang didapatkan (contoh kasus 2)

No	Solusi	Langkah-Langkah	Panjang Solusi
1	Solusi-1	G5, E4, C3, A2	4 langkah

Tampilan *form* simulasi (proses) langkah kuda (posisi awal) untuk mencapai raja (posisi tujuan) terlihat pada gambar 5.7 berikut.



Gambar 5.7 Form simulasi pergerakan langkah kuda

5.2 Pengujian Perangkat Lunak (*Testing*)

Tahap *testing* dilakukan setelah selesai tahap pembuatan dan seluruh data telah dimasukkan. Suatu hal yang tidak kalah penting yaitu aplikasi harus dapat berjalan dengan baik dilingkungan pengguna. Pada tahap pengujian, aplikasi diuji melalui pengujian *blackbox*. Pengujian dengan menggunakan metode *blackbox* yaitu pengujian yang dilakukan untuk antarmuka aplikasi (perangkat lunak), pengujian ini dilakukan untuk memperlihatkan bahwa fungsi-fungsi bekerja dengan baik dalam artian masukan diterima dengan benar dan keluaran yang dihasilkan benar-benar tepat, pengintegrasian eksternal data dapat berjalan dengan baik.

5.2.1 Pengujian Tampilan

5.2.1.1 Pengujian Halaman Input

Pengujian *form* menu *input* pada prekondisi layar menu input dapat terlihat pada tabel berikut.

Tabel 5.3 Tabel Pengujian Halaman Input

No	Deskripsi	Prekondisi	Prosedur pengujian	Masukan	Keluaran yang diharapkan	Hasil yang Didapat	Kesimpulan
1.	Pengujian Tampilan Menu Input	Pengujian Tampilan layar menu Input	Klik menu masuk	Klik Menu masuk	Muncul Pada layar Menu input tombol-tombol menu pilihan yaitu menu <i>Drag Image</i> Kuda, <i>Drag Image</i> Pion, <i>Drag Image</i> Raja, <i>Reset</i> , Proses <i>Shortest Path</i> , <i>About</i> , Keluar	Layar yang ditampilkan sesuai dengan yang diharapkan	Berhasil masuk ke layar utama

				<i>Drag image</i> kuda ke posisi papan catur	Tampil posisi <i>image</i> kuda pada papan catur	Layar yang ditampilkan sesuai dengan yang diharapkan	Posisi <i>image</i> kuda berhasil masuk ke areal papan catur
				<i>Drag image</i> pion ke posisi papan catur	Tampil posisi <i>image</i> pion pada papan catur	Layar yang ditampilkan sesuai dengan yang diharapkan	Posisi <i>image</i> pion berhasil masuk ke areal papan catur
				<i>Drag image</i> raja ke posisi papan catur	Tampil posisi <i>image</i> raja pada papan catur	Layar yang ditampilkan sesuai dengan yang diharapkan	Posisi <i>image</i> raja berhasil masuk ke areal papan catur
				Klik menu <i>Reset</i>	Posisi kuda, pion dan raja diatur ulang kembali	Layar yang ditampilkan sesuai dengan yang diharapkan	Posisi <i>image</i> kuda, pion dan raja berhasil diatur ulang
				Klik Menu proses <i>Shortest path</i>	Muncul Layar Tampilan <i>Shortest path</i>	Layar yang ditampilkan sesuai dengan	Berhasil Masuk ke tampilan <i>Shortest path</i>

						yang diharapkan	
				Klik Menu About	Muncul Layar Tampilan About	Layar yang ditampilkan sesuai dengan yang diharapkan	Berhasil Masuk ke tampilan Simulasi
2.	Pengujian keluar aplikasi (<i>Exit</i>)	Hasil tampilan layar menu input sudah tampil	Klik tombol keluar (X)	Klik menu keluar (<i>Exit</i>)	Keluar dari aplikasi	Layar yang ditampilkan sesuai dengan yang diharapkan	Berhasil keluar dari aplikasi

5.2.1.2 Pengujian Halaman Simulasi

Pengujian *form* menu proses pada prekondisi layar menu simulasi dapat terlihat pada tabel berikut.

Tabel 5.4 Tabel Pengujian Halaman Simulasi

No	Deskripsi	Prekondisi	Prosedur pengujian	Masukan	Keluaran yang diharapkan	Hasil yang Didapat	Kesimpulan
1.	Pengujian Tampilan Simulasi	Pengujian Tampilan layar Simulasi	Klik menu <i>Shortest Path</i>	Klik Menu <i>Shortest Path</i>	Muncul Pada layar Tampilan Simulasi dan Menu Pilihan, yaitu menu <i>Play, Reset, Keluar</i>	Layar yang ditampilkan sesuai dengan yang diharapkan	Berhasil masuk ke layar utama

				Klik Menu <i>Play</i>	Simulasi berjalan pada Tampilan Simulasi	Layar yang ditampilkan sesuai dengan yang diharapkan	Berhasil Masuk ke tampilan Simulasi
				Klik Menu <i>Reset</i>	Simulasi untuk langkah <i>shortest path</i> selanjutnya	Layar yang ditampilkan sesuai dengan yang diharapkan	Berhasil Masuk ke tampilan Simulasi
2.	Pengujian keluar aplikasi (<i>Exit</i>)	Hasil tampilan layar menu proses <i>Shortest Path</i> sudah tampil	Klik tombol keluar	Klik menu keluar	Keluar dari Tampilan Simulasi.	Layar yang ditampilkan sesuai dengan yang diharapkan	Berhasil keluar dari Tampilan Simulasi

5.1.2.3 Pengujian Halaman About

Pengujian *form* menu *About* pada prekondisi layar menu *About* dapat terlihat pada tabel berikut.

Tabel 5.5 Tabel Pengujian Halaman *About*

No	Deskripsi	Prekondisi	Prosedur pengujian	Masukan	Keluaran yang diharapkan	Hasil yang Didapat	Kesimpulan
1.	Pengujian Tampilan <i>About</i>	Pengujian Tampilan layar <i>About</i>	Klik menu <i>About</i>	Klik Menu <i>About</i>	Menampilkan Tampilan <i>About</i>	Layar yang ditampilkan sesuai dengan yang diharapkan	Berhasil Masuk Ke Tampilan <i>About</i>
2.	Pengujian keluar aplikasi (<i>Exit</i>)	Hasil tampilan layar menu utama sudah tampil	Klik tombol keluar (X)	Klik menu OK (<i>Exit</i>)	Keluar dari Tampilan <i>About</i>	Layar yang ditampilkan sesuai dengan yang diharapkan	Berhasil keluar dari Tampilan <i>About</i>

5.3 Kesimpulan Pengujian

Dari hasil pengujian *Black Box*, didapatkan hasil bahwa Pengujian berdasarkan *Black Box* ternyata sama dengan Simulasi Pergerakan Langkah Kuda Dalam Papan Catur. Keluaran yang dihasilkan oleh aplikasi ini sesuai dengan yang diharapkan yaitu berupa simulasi pergerakan langkah kuda dengan menggunakan metode *Breadth-First Search* (BFS) yang berfungsi untuk mencari semua pergerakan langkah terpendek (*shortest path*) biji kuda dari posisi awal sampai posisi tujuan (raja) dengan rintangan-rintangan (pion) yang telah diberikan dalam papan catur.

BAB VI

PENUTUP

6.1 Kesimpulan

Setelah menyelesaikan aplikasi simulasi pergerakan biji kuda dalam papan catur, penulis menarik kesimpulan sebagai berikut:

1. Aplikasi menggunakan metode pencarian *Breadth First Search* (BFS). Karena itu, pencarian akan menemukan semua solusi terpendek.
2. Aplikasi merupakan implementasi (penerapan) nyata pohon pencarian dalam memecahkan suatu permasalahan berdasarkan konsep *Artificial Intelligence* (AI).
3. Aplikasi mensimulasikan semua gerakan dari semua solusi yang ditemukan, sehingga memberikan gambaran yang cukup jelas atas solusi-solusi yang dihasilkan.

6.2 Saran

Penulis ingin memberikan beberapa saran yang mungkin dapat membantu dalam pengembangan aplikasi ini yaitu:

1. Aplikasi dapat dikembangkan dengan menambahkan beberapa metode pencarian lain yang terdapat di dalam ruang lingkup *Artificial Intelligence* (AI).
2. Aplikasi dapat dikembangkan dengan menambahkan animasi sewaktu biji kuda digerakkan. Untuk animasi yang lebih baik, aplikasi dapat dibangun dengan menggunakan aplikasi *Macromedia Flash*.
3. Aplikasi dapat dikembangkan dengan menambahkan biji catur yang lain.

DAFTAR PUSTAKA

- Bey Magethi, **Pedoman Bermain Catur**. Penerbit Pionir Jaya. Agustus 2009
- Erdi Wijaya, **SPK Untuk Penjadwalan Kuliah Dengan Teknik *Breadth First Search* (studi kasus Jurusan TIF UIN SUSKA RIAU)**. Universitas Islam Negeri Sultan Sarif Kasim. 2005
- Ira rasyidah, **Analisa Perbandingan Algoritma *Breadth First Search* (BFS) Dan Algoritma *Least Cost Search* (LCS) (Studi Kasus: Game Of 8)**. Universitas Islam Negeri Sultan Sarif Kasim. 2007
- Kusumadewi.S, **Artificial Intelligence (Teknik dan Aplikasinya)**, Edisi 2, Penerbit Graha Ilmu, 2002.
- Sandi.S, **Artificial Intelegencia**, Andi Offset Yogyakarta, 1993.
- Suyoto, **Intelegensi Buatan Teori Dan Pemograman**. Penerbit Gava Media. 2004
- http://id.wikipedia.org/wiki/Catur#Arti_istilah_catur. Diakses tanggal 21 Juli 2011
- <http://omkicau.com/2010/05/04/aneka-suara-binatang-liar-binatang-gurun-dan-suara-burung/> Diakses tanggal 13 Juni 2011
- <http://www.google.co.id/picture-white-horse-chess-type-icon-file>. Diakses tanggal 13 juni 2011